# COMPUTATION OF PROBABILISTIC RELATIONSHIP BETWEEN CONCEPTS AND THEIR ATTRIBUTES USING A STATISTICAL ANALYSIS OF JAPANESE CORPORA

*Yoshitaka Kameya and Taisuke Sato*

kameya@mi.cs.titech.ac.jp sato@mi.cs.titech.ac.jp
Dept. of Computer Science, Graduate School of Information Science and Engineering,
Tokyo Institute of Technology
2-12-1 Oookayama, Meguro-ku, Tokyo 152-8552, Japan

## ABSTRACT

In this paper, we describe Semantic Aggregate Model (SAM), a generative probability model for word co-occurrences. We also reformulate its statistical learning algorithm, and present an experimental result using a Japanese corpus. The result shows that SAM effectively extracts relevant words to some aspects of our concepts.

## 1. INTRODUCTION

In the last decade, with the spread of world wide web and large-scaled linguistic corpora, much work for text analysis, such as corpus linguistics, information retrieval, text categorization, and information extraction, has been explored. In text analysis, probability models are often adopted for the reasons such as robustness to the uncertainties in text, comprehensibility in understanding the results of the task, and so on. For example, naive Bayes models are one of the most popular text classifiers.

Semantic Aggregate Model (SAM) [1] is a generative probability model for word co-occurrences, in which it is considered that a co-occurrence of two words is arisen at the mediation by some concept we implicitly have. On the other hand, in Iwayama et al.'s computational model for understanding metaphors [2], several relevant words form an attribute (or a property) of our concepts. So, by using SAM, there is a chance to find a probabilistic relationship between our concepts and their attributes.

In this paper, we first describe SAM in detail, and then reformulate the Expectation-Maximization (EM) algorithm [3] for SAM, in both cases of maximum likelihood (ML) estimation and maximum a posteriori (MAP) estimation. We also show an experimental result using a word co-occurrence data of considerably large size, which are extracted from a Japanese corpus. This result presents an interesting perspective that SAM provides us an effective way to find relevant words for some attributes of our concepts.

This work is supported in part by the 21st Century COE Program "Framework for Systematization and Application of Large-scale Knowledge Resources."

## 2. SEMANTIC AGGREGATE MODEL

Formally, SAM is described as follows. First, $\mathcal{C}$ is a set of possible context classes.[1] We also define a vocabulary $\mathcal{V}$ as a set of possible words. Then, $P(c, w, w')$ is a joint probability that two words $w$ and $w'$ ($w, w' \in \mathcal{V}$) occur under some context class $c \in \mathcal{C}$. It is assumed, in a generative manner, that we first choose some context $c$, and then, according to $c$, choose words $w$ and $w'$ independently (i.e. $w$ and $w'$ are conditionally independent given $c$), and from this assumption, $P(c, w, w')$ can be factorized as follows:

$$P(c, w, w') = P(c)P(w|c)P(w'|c). \tag{1}$$

Basic probabilities $P(c)$ and $P(w|c)$ can be seen as parameters of SAM. The assumption we made above is graphically shown as a Bayesian network in Fig. 1. From the view point of statistical modeling, SAM is a special case of discrete mixture models or naive Bayes models, but there is a difference that we use a common parameter set for $P(w|c)$ and $P(w'|c)$.

If we know all parameters in SAM, for each word $w$, we can compute the membership distribution $P(c|w)$ over $\mathcal{C}$:

$$P(c|w) = \frac{P(c)P(w|c)}{P(w)} = \frac{P(c)P(w|c)}{\sum_c P(c)P(w|c)}. \tag{2}$$

This distribution indicates how often the context $c$ has been occurred when we find the word $w$. So, in terms of such an underlying context, the membership distribution may allow us to capture some conceptual characteristic of the word $w$. From this intuition, the similarity between words $w$ and $w'$ can be measured by $\delta(w, w') = e^{-D(w||w')}$, where

[1] A context class may correspond to an attribute of our concepts, which is introduced in Iwayama et al.'s model.

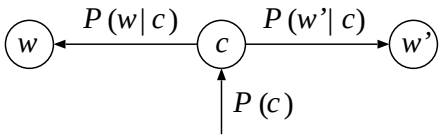


**Fig. 1**. Graphical representation of Semantic Aggregate Model (excerpted from [1]).

$D(w||w')$ is the Kullback-Liebler divergence: $D(w||w') = \sum_c P(c|w) \log \frac{P(c|w)}{P(c|w')}$.

## 3. THE EM ALGORITHM FOR SAM

Before computing the statistical measures above, we need to estimate the parameters $P(c)$ and $P(w|c)$ from a corpus at hand. In observation, however, we cannot see the underlying context $c$ for a word co-occurrence (i.e. $c$ is hidden). This implies that, from a corpus, we only know $N(w, w')$, the number of co-occurrences of $w$ and $w'$.[2] In many of such situations, we can use the Expectation-Maximization algorithm for ML estimation, instead of the so-called relative frequency method. In ML estimation, we attempt to find parameters that maximize the log-likelihood:

$$L = \sum_{w,w' \in \mathcal{V}} N(w, w') \log P(w, w') \ , \qquad (3)$$

where $P(w, w')$ is the probability of a word co-occurrence, and is computed from the model:

$$\begin{aligned} P(w, w') &= \sum_{c \in \mathcal{C}} P(c, w, w') && (4) \\ &= \sum_c P(c) P(w|c) P(w'|c) \ . && (5) \end{aligned}$$

The EM algorithm for SAM is specified in Fig. 2. In the algorithm, the parameters of SAM are iteratively updated until the (log-)likelihood converges. After convergence, we consider the last updated parameters to be an ML estimate which we would like to find. In E-step, we compute two types of expectations $E[c]$ and $E[w|c]$ under the current parameters. The former is an expected count of a context $c$ being occurred, and the latter is an expected count of a word $w$ being occurred when the context $c$ is occurred. In M-step, we update the parameters using these expected counts. $N$ is the total number of occurrences, that is, $N = \sum_{w,w'} N(w, w')$. In Appendix A, we briefly describe the derivation of the EM algorithm for SAM.[3]

[2] $N(w, w')$ should be obtained with no duplications. For example, $N(w, w')$ is the number of times that $w$ occurs in advance of $w'$ in the text.

[3] In the original description [1] of M-step, the constants $1/N$ and $1/2$ are omitted. Indeed these constants are negligible in the computation of E-step or the membership distribution (Eq. 2), but when the EM algorithm is extended to the case of MAP estimation, they cannot be ignored (See Equations 8 and 9).

1. Randomly initialize the parameters $P(c)$ and $P(w|c)$.
2. Iterate the following two steps alternately until $L$ converges (i.e. the difference between the current $L$ and the last one is less than $\varepsilon$):

**E(xpectation)-step:**

$$\begin{aligned} P(c|w, w') &:= \frac{P(w|c)P(w'|c)P(c)}{\sum_c P(w|c)P(w'|c)P(c)} \\ E[c] &:= \sum_{w,w'} N(w, w') P(c|w, w') \\ E[w|c] &:= \sum_{w'} N(w, w') P(c|w, w') \\ &\quad + \sum_{w''} N(w'', w) P(c|w'', w) \end{aligned}$$

**M(aximization)-step:**

$$\begin{aligned} P(c) &:= E[c]/N \\ P(w|c) &:= \tfrac{1}{2} E[w|c]/E[c] \end{aligned}$$

**Fig. 2**. The EM algorithm for SAM

When the number of parameters are increased compared to the size of data, the problem of data sparseness (or zero-frequency) arises. To avoid this problem, from a Bayesian point of view, we adopt MAP estimation instead of ML estimation. Let $\theta$ be the vector of parameters in SAM, and $\mathcal{D}$ be a training data from which we obtain $N(w, w')$. We also introduce a prior distribution $P(\theta)$ for $\theta$, and consider the likelihood $L$ as $P(\mathcal{D}|\theta)$, the distribution of $\mathcal{D}$ given the parameter $\theta$. Then, from Bayes's theorem, it is easy to see that a posteriori distribution $P(\theta|\mathcal{D})$, the probability distribution of $\theta$ given $\mathcal{D}$, is computed as:

$$P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)P(\theta)}{\int P(\mathcal{D}|\theta)P(\theta)d\theta} \propto P(\mathcal{D}|\theta)P(\theta). \qquad (6)$$

MAP estimation is to find the parameters $\theta$ that maximizes a posteriori distribution $P(\theta|\mathcal{D})$. We assume here that $P(\theta)$ follows the Dirichlet distribution:

$$P(\theta) = \gamma \prod_{c \in \mathcal{C}} \{ P(c)^{\alpha} \prod_{w \in \mathcal{V}} P(w|c)^{\alpha'} \}, \qquad (7)$$

where $\gamma$ is a normalizing constant, and $\alpha$, $\alpha'$ are the hyper-parameters of this Dirichlet distribution. From these settings, we obtain the EM algorithm for MAP estimation, by replacing M-step's formulas in Fig. 2 with the followings:[4]

$$\begin{aligned} P(c) &:= \frac{E[c] + \alpha}{N + \alpha|\mathcal{C}|} && (8) \\ P(w|c) &:= \frac{E[w|c] + \alpha'}{2E[c] + \alpha'|\mathcal{V}|} && (9) \end{aligned}$$

[4] In iteration of E-step and M-step, we should check the convergence of a posteriori distribution $P(\theta|\mathcal{D})$, not the likelihood $P(\mathcal{D}|\theta)$.

| $c_1$ | | | | $c_2$ | | | | $c_3$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| adj. $w$ | $P(c_1\|w)$ | noun $w'$ | $P(c_1\|w')$ | adj. $w$ | $P(c_2\|w)$ | noun $w'$ | $P(c_2\|w')$ | adj. $w$ | $P(c_3\|w)$ | noun $w'$ | $P(c_3\|w')$ |
| deliberate | 0.99530 | stance | 0.73869 | beautiful | 0.95468 | tune | 0.74472 | large | 0.98943 | site | 0.87093 |
| clear | 0.97878 | reply | 0.71988 | brave | 0.90716 | harmony | 0.67884 | vast | 0.97669 | range | 0.75925 |
| positive | 0.90652 | statement | 0.68155 | light | 0.90443 | medieval time | 0.64022 | narrow | 0.92767 | view | 0.73006 |
| strong | 0.88882 | response | 0.68151 | elegant | 0.88915 | melody | 0.63640 | gloomy | 0.89550 | stadium | 0.69015 |
| flexible | 0.88234 | attitude | 0.66893 | plain | 0.88651 | dance | 0.58823 | flat | 0.85147 | national land | 0.62448 |

**Table 1**. The membership distribution under the estimated parameters.

It should be noticed that $\alpha$ and $\alpha'$ can be considered as the counts 'by default,' which are added unconditionally to the expected counts $E[c]$ and $E[w|c]$, respectively. Thus we can prevent the parameters from being estimated to zero, even for sparse data.

## 4. EXPERIMENTAL RESULT

The word co-occurrence data used in the experiment[5] is extracted from Mainichi newspaper 1993–2002. In extraction, CaboCha, a Japanese dependency structure analyzer, is used to find syntactic dependency pairs, such as adjective-noun pairs. Each syntactic dependency pair is then considered as a word co-occurrence.

Table 1 shows the membership distribution $P(c|w)$ under the parameters $P(c)$ and $P(w|c)$, which are estimated from word co-occurrence data of adjective-noun pairs. In the co-occurrence data, the total number of words (adjectives and nouns) and co-occurrences are 17,453 and 458,970, respectively. The number of context classes is fixed to 50, and constants $\alpha$, $\alpha'$ and $\varepsilon$ in the EM algorithm are set to $0.1$, $0.1$ and $10^{-6}$, respectively.

Table 1 picks up adjectives and nouns which have top 5 highest membership probabilities for three typical context classes $c_1$, $c_2$ and $c_3$. Note that, for every co-occurrence pair $\langle w, w' \rangle$, we fix $w$ as an adjective, and $w'$ as a noun.[6] From Eq. 2, for a fixed context class $c$, the magnitude of $P(c|w)$ indicates the significance of $P(w|c)$ compared to $P(w)$ (the unconditional or 'averaged' distribution for $w$, since $P(w) = \sum_c P(c)P(w|c)$). Hence we can say each word in Table 1 is closely relevant to the corresponding context class. According to our usual readings for words, it seems that the context classes $c_1$, $c_2$ and $c_3$ correspond to 'attitude,' 'beauty' and 'vastness,' respectively. Several related words are also cleanly extracted for the other context classes.

[5]The experiment we described here is conducted by Asuka Terai and Masanori Nakagawa (Tokyo Institute of Technology).

[6]Especially for such co-occurrence data, we can consider another model [4], in which we have two distinct vocabularies $\mathcal{V}_1$ and $\mathcal{V}_2$ (e.g. adjectives and nouns), and probabilities $P(w|c)$ and $P(w'|c)$ have individual parameter sets such that $\sum_{w \in \mathcal{V}_1} P(w|c) = \sum_{w' \in \mathcal{V}_2} P(w'|c) = 1$. [5] adopts this model in the experiments, and so has slightly different results from the one in this paper.

## 5. CONCLUSION AND RELATED WORK

In this paper, we described Semantic Aggregate Model (SAM) in detail, and reformulated its EM algorithm. SAM is a comprehensible model for word co-occurrences, and the experimental result shows that SAM cleanly extracts related words for some aspects of our concepts. The result is also expected to be used in further analyses of our concepts.

The problem in the EM algorithm is that it can only find local ML (or MAP) estimates, and so it is needed to adopt some promising methods including deterministic annealing EM (DAEM) algorithm [6] or split-merge EM (SMEM) algorithm [7] to avoid being trapped in 'bad' local estimates. There is also a question of how to determine an appropriate number of context classes. This can be seen as a problem of model selection, which is intensively discussed in the last three decades.

## 6. REFERENCES


[1] D. Mochihashi and Y. Matsumoto, "Probabilistic representation of meanings," in *IPSJ SIG Note NL-147*, 2002, pp. 77–84, In Japanese.

[2] M. Iwayama, T. Tokunaga, and T. Tanaka, "The role of salience in understanding metaphors," *Transactions of the Japanese Society for Artificial Intelligence*, vol. 6, no. 5, pp. 674–681, 1991, In Japanese.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. B39, pp. 1–38, 1977.

[4] F. Pereira, N. Tishby, and L. Lee, "Distributional clustering of English words," in *Proc. of the 31st Annual Meeting of the Association for Computational Linguistics*, 1993, pp. 183–190.

[5] A. Terai, "From computation to mind: examining the psychological validity of a computational model of metaphor understanding — targeting more human-like systems," in *Proc. of Symposium on Large-scale Knowledge Resources: LKR2005*, 2005, this volume.

[6] N. Ueda and R. Nakano, "Deterministic annealing EM algorithm," *Neural Networks*, vol. 11, no. 2, pp. 271–282, 1998.

[7] N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton, "SMEM algorithm for mixture models," *Neural Computation*, vol. 12, no. 9, pp. 2109–2128, 2000.

## A. DERIVATION OF THE EM ALGORITHM FOR SEMANTIC AGGREGATE MODEL

We first describe a general form of the EM algorithm for ML estimation. Let $\mathcal{Y}$ be observed data and $\mathcal{X}$ be missing data correspond to $\mathcal{Y}$.[7] Also Q function is defined as follows:

$$\begin{aligned} Q(\theta', \theta) &\stackrel{\text{def}}{=} E\{\log P(\mathcal{X}, \mathcal{Y}|\theta')|\mathcal{Y}, \theta\} \\ &= \textstyle\sum_{\mathcal{X}} P(\mathcal{X}|\mathcal{Y}, \theta) \log P(\mathcal{X}, \mathcal{Y}|\theta'). \end{aligned} \quad (10)$$

Then, for $t = 0, 1, 2, \ldots$, E-step of the EM algorithm is to compute $Q(\theta, \theta^{(t)})$, and M-step is to compute $\theta^{(t+1)} = \arg\max_\theta Q(\theta, \theta^{(t)})$, where $\theta^{(0)}$ and $\theta^{(t)}$ are the initial parameters and the parameters obtained after $t$-th iteration ($t = 1, 2, \ldots$), respectively. It can be shown that $Q(\theta', \theta) \geq Q(\theta, \theta)$ implies $P(\mathcal{Y}|\theta') \geq P(\mathcal{Y}|\theta)$, and from M-step, we can say $Q(\theta^{(t+1)}, \theta^{(t)}) \geq Q(\theta^{(t)}, \theta^{(t)})$. Hence it is ensured in general that the likelihood $P(\mathcal{Y}|\theta)$ monotonically increases while updating $\theta$ by E-step and M-step.

For the EM algorithm specialized for SAM, we consider the observed data $\mathcal{Y}$ (which corresponds to $\mathcal{D}$ in Section 3) as a sequence of $N$ independent word co-occurrences $\langle w_i, w_i' \rangle$, and missing data $\mathcal{X}$ is a sequence of the corresponding context classes $c_i$. Then the specialized form of Q function is obtained as:

$$\begin{aligned} Q(\theta', \theta) &= \sum_{i=1}^{N} \sum_{c_i \in \mathcal{C}} P(c_i|w_i, w_i', \theta) \log P(c_i, w_i, w_i'|\theta') \\ &= \textstyle\sum_{w,w'} N(w, w') \cdot \\ &\quad \textstyle\sum_c P(c|w, w, \theta) \log P(c, w, w|\theta') \end{aligned} \quad (11)$$

(the proof is omitted). Note here that $N(w, w')$ is defined as the number of occurrences of $\langle w, w' \rangle$ in $\mathcal{Y}$, and $N = \sum_{w,w'} N(w, w')$. We hereafter abbreviate the parameters $P(c)$ and $P(w|c)$ as $\theta_c$ and $\theta_{w|c}$ respectively. Following the general description of EM, given $\theta = \theta^{(t)}$, our goal is now to find $\theta' = \theta^{(t+1)}$ that maximizes $Q(\theta', \theta)$. This can be regarded as a constrained optimization problem, where the constraints are that $\sum_c \theta_c' = 1$ and $\sum_w \theta_{w|c}' = 1$ ($c \in \mathcal{C}$), and so we will use the Lagrange multiplier method. Let us consider the function:

$$F(\theta') = Q(\theta', \theta) - \lambda(\textstyle\sum_c \theta_c' - 1) - \sum_c \lambda_c (\sum_w \theta_{w|c}' - 1),$$

where $\lambda$ and $\lambda_c$ are arbitrary constants. Substituting Eq. 11 and $P(c, w, w'|\theta') = \theta_c' \theta_{w|c}' \theta_{w'|c}'$ to $F(\theta')$ above, we have:

$$\begin{aligned} F(\theta') &= \textstyle\sum_{v,v'} N(v, v') \sum_c P(c|v, v', \theta) \cdot \\ &\quad (\log \theta_c' + \log \theta_{v|c}' + \log \theta_{v'|c}') \\ &\quad -\lambda(\textstyle\sum_c \theta_c' - 1) - \sum_c \lambda_c (\sum_v \theta_{v|c}' - 1). \end{aligned}$$

Here, for each $c \in \mathcal{C}$, we obtain $\theta_c' = E[c]/\lambda$ from:

$$\frac{\partial F(\theta')}{\partial \theta_c'} = \frac{1}{\theta_c'} \textstyle\sum_{v,v'} N(v, v') P(c|v, v', \theta) - \lambda = 0$$

and $E[c] \stackrel{\text{def}}{=} \sum_{w,w'} N(w, w') P(c|w, w', \theta)$. Since $\sum_c \theta_c' = 1$, $\lambda = \sum_c E[c]$ holds. It is easy to see $\sum_c E[c] = N$, and hence we finally get:

$$\theta_c' = E[c]/N. \quad (12)$$

On the other hand, for each $c \in \mathcal{C}$ and $w \in \mathcal{V}$, we obtain $\theta_{w|c}' = E[w|c]/\lambda_c$ from:

$$\begin{aligned} \frac{\partial F(\theta')}{\partial \theta_{w|c}'} &= \frac{1}{\theta_{w|c}'} \textstyle\sum_{v'} N(w, v') P(c|w, v', \theta) + \\ &\quad \frac{1}{\theta_{w|c}'} \textstyle\sum_v N(v, w) P(c|v, w, \theta) - \lambda_c = 0 \end{aligned}$$

and

$$\begin{aligned} E[w|c] &\stackrel{\text{def}}{=} \textstyle\sum_{w'} N(w, w') P(c|w, w', \theta) \\ &\quad + \textstyle\sum_{w''} N(w'', w) P(c|w'', w, \theta). \end{aligned}$$

Since $\sum_w \theta_{w|c}' = 1$, $\lambda_c = \sum_w E[w|c]$ holds. It is also easy to see $\sum_w E[w|c] = 2E[c]$, and hence the following is obtained:

$$\theta_{w|c}' = E[w|c]/2E[c]. \quad (13)$$

The EM algorithm in Fig. 2 is now derived by regarding the computation of $E[c]$ and $E[w|c]$ as E-step, and Eqs. 12 and 13 as M-step.

[7] Since we have a missing part $\mathcal{X}$, it is said that the observed data $\mathcal{Y}$ is incomplete, while the pair $(\mathcal{X}, \mathcal{Y})$ is called complete data. Then $P(\mathcal{X}, \mathcal{Y}|\theta)$ is the likelihood of complete data under the parameters $\theta$. Note here that $P(\mathcal{Y}|\theta)$ is the likelihood in an usual sense.